

SVEUČILIŠTE U ZAGREBU
FILOZOFSKI FAKULTET
ODSJEK ZA INFORMACIJSKE I KOMUNIKACIJSKE ZNANOSTI
Ak. god. 2017./ 2018.

Ivan Bradač

CSS Grid raspored

Završni rad

Mentor: dr.sc. Kristina Kocijan, doc.

Zagreb, 2018

Sadržaj

Sažetak	2
1. Uvod	3
2. Osnovni pojmovi vezani uz CSS grid	5
2.1. HTML.....	5
2.2. CSS.....	5
3. Povijest nastanka CSS Grid-a.....	6
3.1. Advanced Layout Module	6
3.2. Microsoft	7
3.3. Standardizacija	9
Sadržaj.....	1
4.1. CSS terminologija	11
4.2. Stvaranje jednostavnog grid-a	12
4.3. Grid područja.....	12
4.4. Unos elemenata u grid.....	13
5. Usporedba CSS Grid-a, CSS Flexbox-a i Bootstrap-a	15
5.1. CSS Grid u usporedbi s CSS Flexbox-om.....	15
5.2.CSS Grid raspored kao zamjena za Bootstrap layout	15
6. Stvaranje rasporeda postojećih web stranica uz pomoć CSS-grid-a.....	18
6.1. Dinamični raspored mrežne stranice po uzoru na YouTube u CSS grid-u.....	18
6.2. Raspored mrežne stranice grada Samobora u CSS grid-u	19
7. Zaključak	21
8. Literatura	22
9. Prilozi	24
9.1. Prilog 1 Kodni blok mrežne stranice po uzoru na Youtube u CSS Grid-u.....	24
9.1.1. HTML.....	24
9.1.2. CSS.....	26
9.2. Prilog 2: Kod rasporeda mrežne stranice grada Samobora u CSS grid-u.....	28
9.2.1. HTML.....	28
9.2.2. CSS.....	29

Sažetak

Predmet rada je prikaz osnovnih pojmova, obilježja i metoda koje se koriste u izradi strukture i rasporeda web stranice pomoću alata iz CSS grid skupine naredbi. Cilj rada je prikazati mogućnosti koje pruža ova novo implementirana vrsta naredbi uz pomoć kojih je izrazito olakšan rad na mrežnim stranicama. Prikazano je postavljanje novih standarda izgleda web stranica. Nakon kratkog uvoda, dano je kratko objašnjenje pojmova koji su vezani za područje CSS grida i Web dizajna. Potom je dan povijesni pregled razvitka CSS grida sve do danas. Objašnjena je osnovna terminologija koja prati CSS grid, koristeći vizualnih prikaza važnijih dijelova kod izgradnje rasporeda mrežne stranice. Učinjena je usporedba CSS grid-a sa alatima izrade rasporeda mrežnih stranica koji se danas najviše koriste. Na kraju su kroz dva primjera repliciranja mrežnih stranice prikazana unaprjeđenja koje CSS grid donosi.

Ključne riječi: *CSS-grid, HTML, mrežne aplikacije, dizajn mrežnih stranica, Bootstrap, raspored*

CSS Grid layout

Abstract

This paper deals with the basic concepts and methods used to create the structure and layout of a website using the CSS Grid command tool. The purpose of this paper is to showcase the features provided by these newly implemented types of commands, which have made it significant easier to work on websites. The setting of new web page standards is shown. After introduction, a brief explanation of terms related to CSS grids and web design is provided. Then a historic review of the CSS grid is given up until today. The basic terminology that follows the CSS grid is explained by visual portrays of the most important parts that build a web page. A comparison between CSS grid and other tools that are most commonly used today is made. In the end, two examples of replication of popular web pages showing the improvements introduced by the CSS grid are given.

Keywords: *CSS-Grid, HTML, Network Applications, Web Site Design, Bootstrap, Schedule*

1. Uvod

Pojavom interneta, mrežne su stranice postepeno počele zahtijevati određeni dizajn i strukturu kako bi se na što lakši način prezentirao sadržaj. Mrežne stranice se u početku prikazuju samo simbolima i tabulatorima (Ruluks, 2014). Dakle, potreba za uređenijim prikazom mrežnih stranica postoji od samoga početka. Potpunim uvođenjem *Cascading Style Sheets* (CSS) 1998. godine napravljen je prvi veliki korak u tome smjeru.

Popularizacijom interneta i razvojem interaktivnih mrežnih stranica, otvara se novi načini prikaza informacija velikom broju korisnika. Mrežne stranice primorane su promijeniti se iz temelja.

Izgledu stranice počinje se posvećivati sve više pažnje. Korisnici traže mrežne stranice koje će zadovoljiti njihove potrebe, a uz to izgledati privlačno. Razvojem suvremenih mrežnih stranica sve se više približavamo, kako izgledom, tako i funkcionalnošću, klasičnim desktop aplikacijama. Brzine koje današnji internet pruža korisnicima preko mrežnih aplikacija postavljaju vrlo visoke zahtjeve u pogledu dinamičnosti, brzine, reakcije na akcije korisnika, i slične aspekte rada kojima se bave programeri i dizajneri mrežnih stranica.

Iz toga proizlazi velika potreba, za novim programskim rješenjima. W3S CSS radna skupina koja je najznačajnija prilikom odobrenja novih standarda unutar CSS-a primorana je postaviti nove standarde. Jedan od prijedloga je i sustav koji uključuje mrežu (engl. *grid*), kako bi se olakšala izrada rasporeda elemenata. Takav sustav ima dugu povijest korištenja u dizajnu, ali njegova implementacija i standardizacija nisu prihvaćene od strane mrežnih preglednika zbog kompleksnosti.

Kao posljedica toga CSS je primoran razvijati se u drugome smjeru. Složenost mrežnih stranica eksponencijalno raste, raspored elemenata također. Programeri i dizajneri morali su tražiti nova rješenja pomoću *JavaScripta*, *Bootstrap-a* ili nekog *Framework-a*.

Vidjevši na *WebCamp-u* 2017. godine predavanje Mortena Rand-Hendriksena o CSS *grid-u* našao sam rješenje za problem rasporeda i izgleda mrežnih stranica. Korištenje CSS *grid-a* značilo je čisti izgled mrežne stranice koristeći samo CSS i *HTML*.

Glavni cilj ovog rada je prikazati povijesni razvitak CSS *grid-a*, objasniti koje nam nove mogućnosti daje, zašto se pojavio tek dvadeset godina nakon pojave samog CSS-a i što to znači za budućnost *web* dizajna.

Kako bih dao bolji pregled ovaj sam rad podijelio u šest cjelina: nakon kratkog uvoda u drugoj cjelini objašnjavam osnovne pojmove koji se trebaju poznavati kako bi se razumjeli teorijske cjeline *CSS grid*-a; u trećoj cjelini ću proći kroz povijest *CSS grid*-a. Od početne ideje Bert Bos-a, pa sve do implementacije u mrežne preglednike. Unutar poglavlja posebno ću spomenuti ulogu *Microsoft*-a u razvitku algoritma na kojem se danas sve temelji. U četvrtoj ću cjelini proći kroz osnove *CSS grid*-a, terminologiju koja je važna za početak stvaranja *grid* rasporeda mrežnih stranica. Objasnit ću koja su svojstva *CSS grid*-a. Detaljnije ću objasniti *grid* retke, *grid* stupce i *grid* polja područja, te kako kroz njih možemo pozicionirati *grid* elemente. Peto poglavlje posvećeno je usporedbi danas najviše korištenih alata za kreiranje rasporeda mrežne stranice, *CSS Flexbox*-a i *Bootstrap*-a s *CSS grid*-om. Navest ću prednosti i nedostatke rada s tim alatima. Šesto poglavlje posvećeno je prikazivanju jednostavnosti stvaranja mrežne stranice koristeći samo svojstva *grid* modula.

2. Osnovni pojmovi vezani uz CSS grid

U ovom poglavlju objasni ću terminologiju koje vežemo s *CSS Grid*-om i *web* dizajnom općenito. Kroz potpoglavlja ću proći kroz pojmove *HTML*-a i *CSS*-a.

2.1.HTML

HTML je skraćenica od *HyperText Markup Language*. *HTML* je jezik za označavanje hipertekstualnih dokumenata. Osnovni je jezik kojim stvaramo i opisujemo mrežni dokument. Dobro oblikovani *HTML* dokument omogućuje web pregledniku prikaz strukture dokumenta. Kako bismo omogućili pregledniku bolji prikaz strukture koristimo naredbe, sve *HTML* naredbe moraju se nalaziti unutar oznake za manje (<) i oznake za više (>). Većina naredbi ima oznaku za svoj početak i oznaku za svoj kraj. Oznaka za kraj je uvijek simbol "/". Svaki *HTML* dokument mora imati određenu strukturu, oznaku početka <HTML> i oznaku kraja </HTML>, zaglavlje <head> </head> unutar kojeg upisujemo opće definicije i tijelo <body> </body> unutar kojeg upisujemo sve ono što želimo da se vidi u pregledniku (W3schools, 2018). *HTML* datoteke su obične tekstualne datoteke s ekstenzijom .htm ili .html, a mogu biti kreirane korištenjem bilo kojeg tekst editora.

2.2.CSS

Cascading Style Sheets je stilski jezik, koji se koristi za opis izgleda dokumenta napisanog pomoću jezika za označavanja (*HTML*). *CSS* nam omogućuje stilske promjene mrežnog dokumenta kako bi postigli veću preglednost i lakše uređivanje. *CSS* možemo pisati u samome mrežnom dokumentu, ali u većini slučajeva stvaramo odvojeni *CSS* dokument. *CSS* sintaksa sastoji se od popisa pravila koja opisuju izgled elemenata *HTML* dokumenta. Osnovna sintaksa *CSS*-a jest sljedeća: selektor {svojstvo: vrijednost;}. Selektor je zapravo oznaka ili jedan od njezinih atributa (najčešće klasa ili id) unutar *HTML* dokumenta kojom točno definiramo nad čemu *CSS* vrši stiliziranje. Svaki selektor mora završavati znakom točka-zarez, čak i kada je u retku deklariran samo jedno svojstvo (W3schools, 2018).

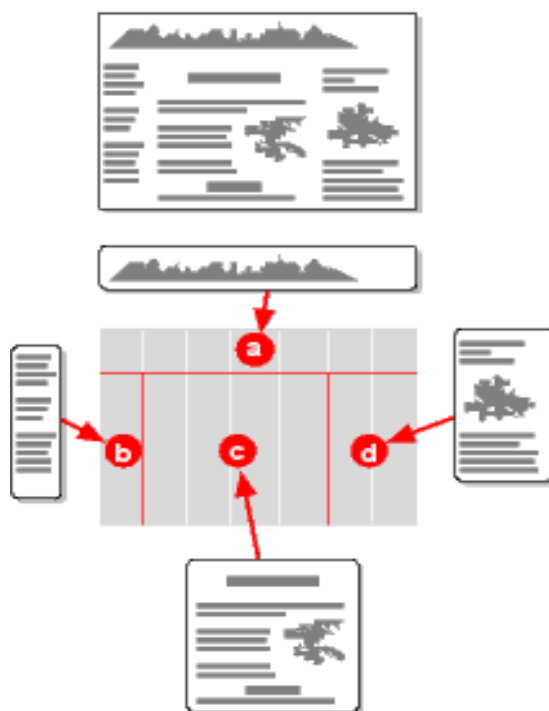
3. Povijest nastanka CSS Grid-a

Poglavlje tri daje povijesni pregled *CSS grid*-a. Kroz potpoglavlja prikazat ću važne događaje koji su doveli do *CSS Grid* rasporeda kakvog poznajemo danas.

3.1. Advanced Layout Module

Raspored pomoću mreže (engl. *grid*) koristi se u svim industrijama prilikom raspoređivanja sadržaja i dizajniranja proizvoda. Mreža je osnova za precizan i uredan izgled .

Kod dizajniranja mrežnih stranica pomoću CSS-a, takva mogućnost nije postojala, zbog nedostatka podrške od strane internet preglednik. Oni su tu ideju smatrali prekompleksnom za implementaciju (Gustafson, 2017). Dr. Bert Bos koji je uz Håkon Wium Liea 1994. predstavio CSS, izrađuje nacрте raspoređivanja elemenata web stranica pomoću mreže. Predaje ih W3S CSS radnoj skupini (*CSS Working Group*) u više navrata. Prvi nacrt koji predstavlja strukturu unutar CSS-a sličnu današnjem izgledu *grid*-a naziva *Advanced Layout Module* (kasnije *CSS Template Layout Module*). U njemu Bos dijeli mrežnu stranicu na ćelije te unutar njih raspoređuje sadržaj u „područja“ (Slika 1).



Slika 1 - Prikaz podjele mrežne stranice prema Dr. Bert Bos-u
Preuzeto s: <https://www.w3.org/TR/2007/WD-css3-layout-20070809/>

Ćelije na koje Bos dijeli stranicu definira novom CSS sintaksom koja prvi put koristi mrežu simbola. Njegov nacrt usprkos podršci dizajnera mrežnih stranica nikad nije implementiran u neki mrežni preglednik. Iako neuspješan, Bos time postavlja temelje iz kojih će se godinama kasnije stvoriti CSS grid.

Primjer Bos-ove zamisli pozicioniranja:

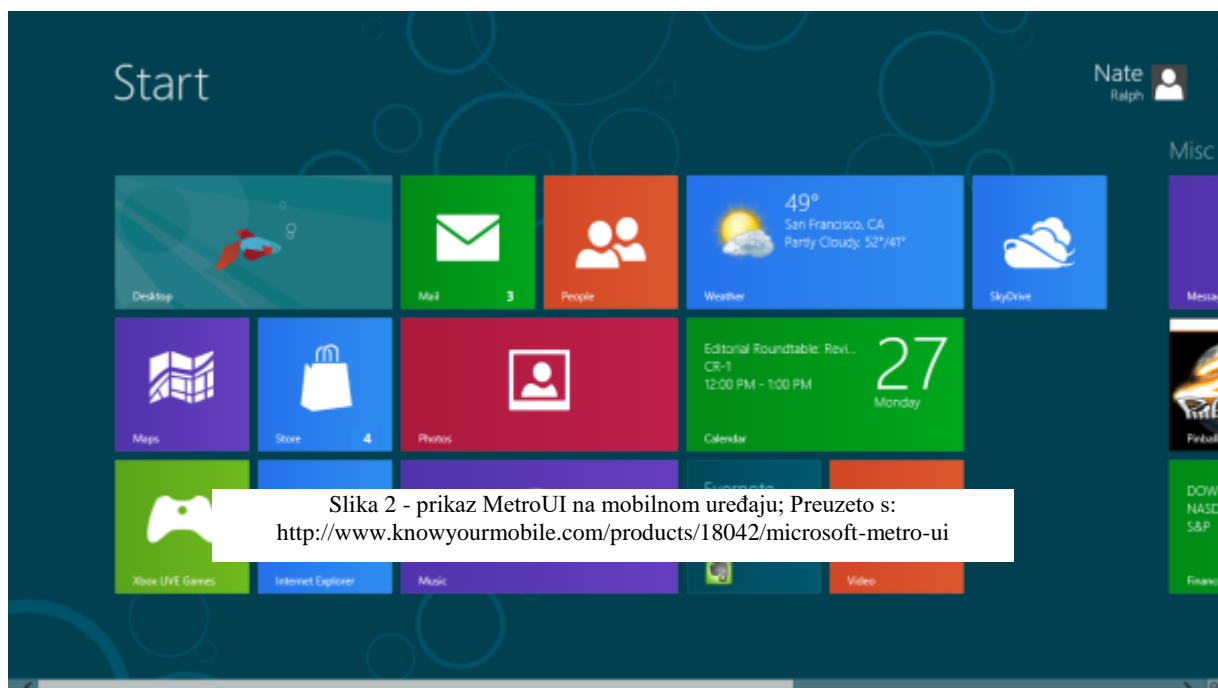
```
<style type="text/css">
dl { display: "ab" "cd" }
#sym1 { position: a }
#lab1 { position: b }
#sym2 { position: c }
#lab2 { position: d }
</style>
```

3.2. Microsoft

Počeci testiranja i implementacije *CSS grid*-a u mrežne preglednike dolaze iz *Microsoft* 2011. godine. Skupina dizajnera i programera predvođena Philipom Cuppom dobiva zadatak ujedinjenja raznih vrsta uređaja koje bi koristile *Microsoft*-ov softver preko oblaka (engl. *clouda*). Vidjevši da velika većina programera, zbog malog broja potrebnih rezolucija (ukupno četiri) stvara aplikacije prvenstveno za *Apple iPhone* i *iPad*. Cupp je primoran stvoriti fleksibilno programski rješenje kojim bi se mogle prikazati aplikacije na bilo kojoj rezoluciji i veličini ekrana.

Microsoft 2011. godine najavljuje svoj novi operacijski sustav *Windows 8* uz kojega planiraju omogućiti izradu mrežnih aplikacija. To omogućuje novom timu s voditeljima Alexom Mogilevskyim i Markusom Mielkeom, kojima se pridružuje i Cupp da počnu od nule. U svojem radu primjećuju nedostatak alata za precizno stvaranje mrežnih aplikaciji. Njihova je zamisao stvoriti jednostavan alat kojim je moguće strukturiranje i raspoređivanja mrežnih sadržaja po uzoru na Bos-ove nacрте (Gustafson, 2017)

Uz najavu izlaska *Microsoft Windows 8* i *Internet Explorer* 10 2011. godine javnosti je predstavljena i Microsoftova inačica *grid* rasporeda koja će se moći koristiti u *Internet Exploreru*. *Microsoft*-ov CSS *grid* koristi –ms- prefiks (Grid Layout: W3C Working Draft7).



Mrežna arhitektura vidljiva je i u samome *MetroUI* korisničkom sučelju. Aplikacije se žele približiti korisnicima na isti način neovisno o uređaju (Slika 2., Slika 3)

Nakon uspješne primjene mrežne arhitekture u praksi putem *Windows 8* i *Internet Explorer-a* 10 Mogilevsky, Mielke i Cupp 2012. godine šalju nacrt „*Grid Layout spec*“ CSS radnoj skupini. Prvi je to puta da se uz nacrt prikazuje implementacija koju je moguće

evaluirati. Rachel Andrew jedna je od eksperta pozvanih na CSS radnu skupinu, odmah je bila oduševljena *CSS gridom* i mogućnostima koje pruža. Odmah po implementaciji u *Internet Explorer*, Andrew kreće u stvaranje primjera kako bi ostalim web dizajnerima približila mogućnosti grida.(Andrew, 2012)

3.3.Standardizacija

Microsoft-ov grid i njegova implementacija u preglednik bio je početni korak. Dizajniranje rasporeda web stranica pomoću mreže (engl. *grid*) trebalo je još samo standardizirati. Nakon prvih povratnih informacija CSS radna skupina i *Microsoft* zajedno rade na poboljšanju algoritma. Peter Linss dopredsjednik CSS radne skupina došao je na ideju uvrštavanja linija unutar grid modula.

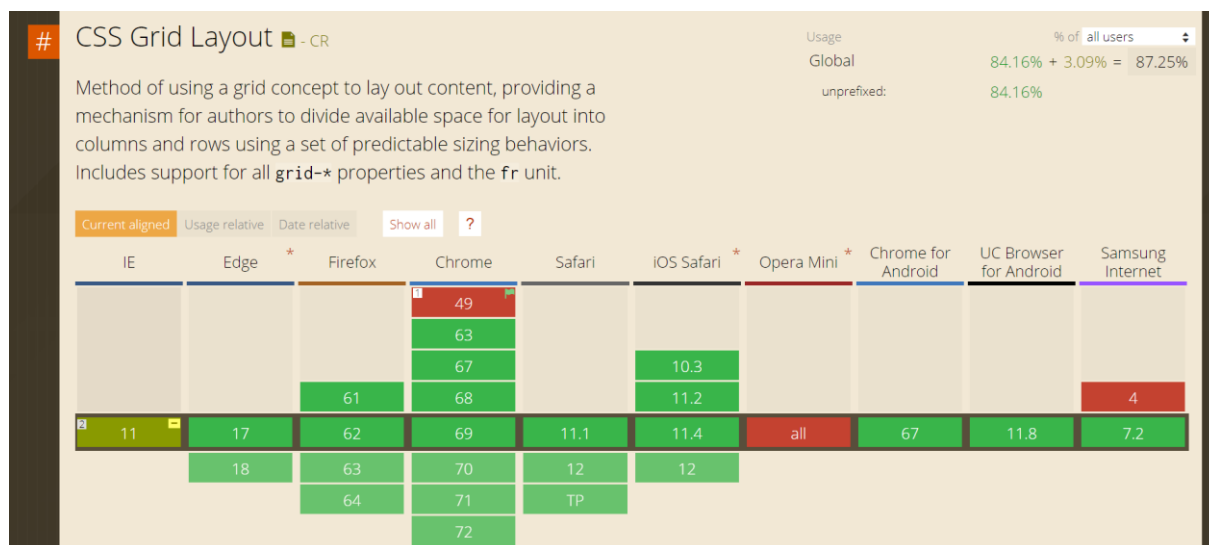
W3C CSS radna skupina imala je zadaću naći kompromis između Bosova *Advanced Layout Module* nacрта, ideja *Microsoftovih* programera i dizajnera, te zamisli Petera Linssa. Zaključak (Gustafson, 2017) je da za implementaciju CSS grid treba biti:

1. **opširan**, kako bi dizajneri mogli stvoriti sve što poželes bez ograničenja
2. **robustan**, kako ne bi došlo do raspada rasporeda web stranice ili nestanka sadržaja prilikom rukovanja krajnjeg korisnika s web stranicom
3. **snažan**, tako da algoritam CSS grida ne usporava rad preglednika prilikom promjene veličine prozora preglednika ili dinamičnog prikazivanja sadržaja

Nakon pronalaska rješenja kako ispuniti zadane ciljeve 2012. godine CSS radna skupina trebala je dvije samostalne, interoperabilne implementacije *CSS grida*. Prva je implementacija već postojala od strane *Microsofta*, druga implementacija došla je 2013. godine od strane *open-source* tvrtke *Igalia*. Ona na zahtjev *Bloomberg-a* uvodi *CSS grid* u *Blink* i *WebKit* alate za izradu web stranica. Njihov je doprinos bio velik zbog mogućnosti njihovih *web* dizajnera i programera da daju neovisne povratne informacije o poboljšanjima(Andrew, 2015). Finalni nacrt time postaje predloženom preporukom (engl. *proposed recommendation*).

Rachel Andrew povećava popularnosti *CSS grid-a* svojim primjerima. Uz nju se još može spomenuti Jen Simmons dizajnerica *Mozilla*. Ona je pokrenula stranicu Labs na kojoj je omogućila pristup demonstracijama mogućnost *CSS grid-a* koje je ona stvorila.

U siječnju 2017. godine Google *Chrom* prvi uvodi *CSS* u svoj Chromium 56 preglednik. Unutar jednog mjeseca svi veći web preglednici počeli su podržavati *grid* standard. Danas 87.25% svih mrežnih preglednika podržava *CSS grid*, što je vidljivo na slici 4.



Slika 4 Prikaz podrške CSS grida od strane najvećih mrežnih preglednika Preuzeto s <https://caniuse.com/#search=css%20grid>

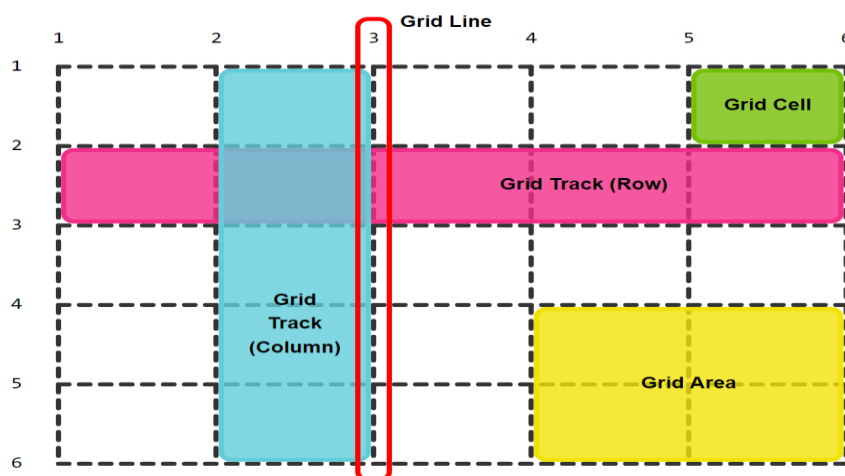
4. Osnove CSS Grid rasporeda

U ovom ću poglavlju objasniti građu svakog *grid* rasporeda. Kroz potpoglavlja ću objasniti dijelove koje je potrebno definirati prije početka rada s CSS grid-om.

4.1. CSS terminologija

CSS grid raspored započinje odabirom prikaza grida pomoću CSS svojstva *display*. Svojstvo *display* zapravo definira spremnik (engl. *container*) unutar kojega želimo stvoriti *grid*. Osim *grid* spremnika-a sastavne komponente *grid*-a (slika 5) prema Meyeru (2016):

1. **grid item** je izravni nasljednik grid spremnika (samo onaj na prvoj razini) koji automatski postaje grid stavka.
2. **grid line** horizontalna (red) ili vertikalna (kolona) linija koja razdvaja grid u sekcije.
3. **grid cell** označava ćeliju unutar grida.
4. **grid area** (područje) je pravokutno područje između određenog broja grid linija.
5. **grid track** je prostor između dvije susjedne grid linije. Možemo ih gledati horizontalno kao red ili vertikalno kao kolonu



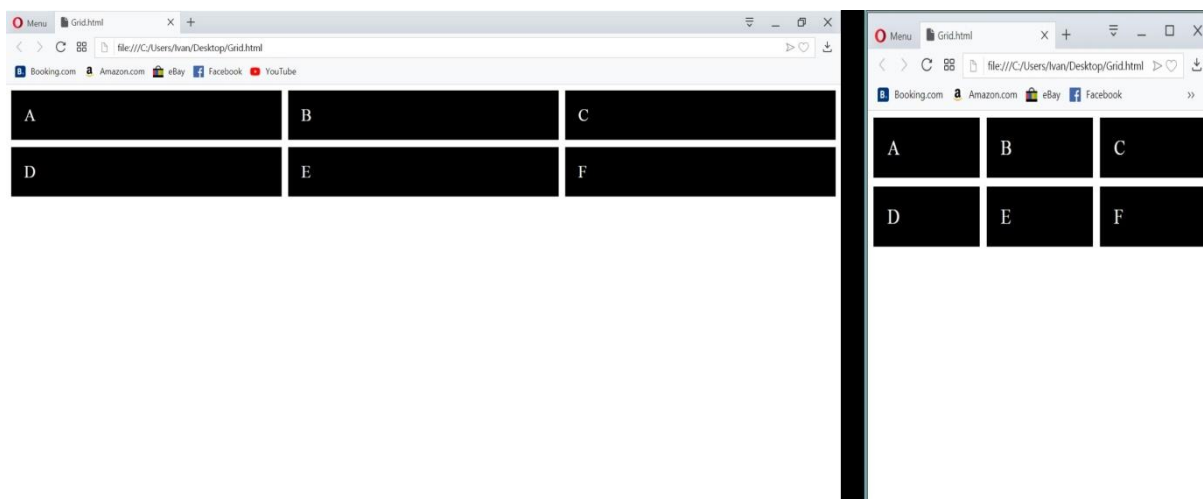
Slika 5- terminologija CSS-grida; Preuzeto s: <https://webkit.org/blog/7434/css-grid-layout-a-new-layout-module-for-the-web/>

4.2. Stvaranje jednostavnog grid-a

Nakon što smo prošli svu terminologiju koja nam je potrebna za razumijevanje *grid*-a možemo početi s definiranjem same njegove strukture. Prvo definiramo grid container uz pomoć svojstva *display*, zatim definiramo svojstva *grid-template-rows* i *grid-template-columns*. *Grid-template-rows* određuje koliko ćemo redaka imati u gridu, a *grid-template-columns* određuje broj stupaca koje će grid imati. Navedenim svojstvima potrebno je dodati mjeru veličine stupca ili retka CSS grid-a.

Mjerila za određivanje veličine stupaca i redaka CSS *grid*-a mogu biti izražene različito. Najčešće koristimo auto (automatsko popunjavanje), u obliku broja piksela (oznaka px nakon broja), ili u obliku frakcije(fr nakon broja).

Auto mjerilo se prilagođava ostalim zadanim mjerilima. Korištenjem piksela dobivamo grid ćelije koje su fiksne dimenzije neovisno o tome koliko se promjeni rezolucija. Frakcija je nova mjera koja se javlja u CSS3 posebno za CSS *grid*. Jedinica frakcije, predstavlja dijelove raspoloživog prostora u stupcu ili retku. Ako želimo stranicu napraviti tako da se promjenom rezolucije preglednika ne mijenja njezin raspored u gridu (slika 6) koristimo upravo frakcije(Meyer,2016).



Slika 6 - prikaz rezultata korištenja frakcija na različitim rezolucijama osobnog računala(lijevo) i mobilni uređaj (desno)

4.3. Grid područja

Koristeći *grid-template-columns* i *grid-template-rows* mogli smo točno definirati broj redaka i stupaca od kojih želimo da se naš mrežni dokument sastoji. Još jedan način kako to napraviti je korištenje svojstva *grid-template-area*. Po uzoru na Bol-ov nacrt *Advanced*

Layout Module, ovim svojstvom definiramo pozicije pojedinih područja mreže stranice. Kako bi to izveli *HTML* oznakama pridodajemo *CSS* svojstvo ***grid-area***. Zatim vrijednost svojstva *grid-area* ubacujemo u *grid-area-template*:

```
#grid {display: grid;
      grid-template-areas:
        "header header header header"
        "leftside content content rightside"
        "leftside footer footer footer";
      @page > head {
        grid-area: header;
        background-color: #8ca0ff;
      }
```

HTML oznaci `<head>` kao što je vidljivo u kodnom bloku iznad dodjeljujemo *grid-area* svojstvo „header“. Ako pogledamo u *grid-template* vidjet ćemo da se header pojavljuje samo u prvom redu i to u sva četiri stupca. *CSS grid* će sve što se nalazi u *HTML* oznaci `head` prebaciti na taj položaj u gridu (Meyer, 2016).

4.4. Unos elemenata u grid

Osim *grid* područja elementi se mogu unositi pomoću svojstva kraja i početka redova, stupaca:

1. `grid-row-start <number> | <name> | span <number> | span <name> | auto`
2. `grid-row-end <number> | <name> | span <number> | span <name> | auto`
3. `grid-column-start <number> | <name> | span <number> | span <name> | auto`
4. `grid-column-end <number> | <name> | span <number> | span <name> | auto`

Svojstvima koje smo gore naveli dodajemo indeks, naziv ili raspon ćelija u kojim želimo da element počinje i završava (House, 2018).

grid-column: 1; grid-row: 1;	grid-column: 2; grid-row: 1;	grid-column: 3; grid-row: 1;
grid-column: 1; grid-row: 2;	grid-column: 2; grid-row: 2;	grid-column: 3; grid-row: 2;

Slika 7 -Prikaz indeksa grid polja : Preuzeto s <https://medialoot.com/blog/a-beginners-guide-to-css-grid-layout/>

Slika 7 prikazuje nam kako se indeksiraju redci i stupci unutar *CSS- grid*-a. Pomoću indeksa možemo lako definirati početak i raspon grid područja. Ako grid područje zahvaća veliki broj grid ćelija ovo svojstvo smanjuje potrebu za velikim brojem podataka o ćelijama koji su potrebni kod *grid-area-template* svojstva.

5. Usporedba CSS Grid-a, CSS Flexbox-a i Bootstrap-a

Ovo poglavlje posvećeno je usporedbi mogućnosti *CSS grid*-a i alata koji se danas najviše koriste upravo za raspored elemenata mrežne stranice. Predstaviti ću sve prednosti i mane tih alata, te pokazati ponavljaju li se ti problemi i u *CSS grid*-u.

5.1. CSS Grid u usporedbi s CSS Flexbox-om

CSS Flexbox postao je standard W3S *CSS* radne skupine već 2012. godine. Sastavni je dio svake mrežne stranice. Postao je iznimno popularan među programerima posljednjih nekoliko godina. To ne čudi, jer nam je olakšao stvaranje dinamičkih izgleda i poravnavanje sadržaja unutar spremnika.

Flexbox je ograničen na jednodimenzionalni dizajn. Moguća je manipulacija samo horizontalnog ili samo vertikalnog izgleda elemenata *Flexbox*-a.

S *CSS grid*-om možete raditi na dvodimenzionalni način, što znači da možemo istovremeno raditi sa stupcima i redovima (Soares, 2018).

Flexbox je namijenjen radu sa sadržajem. Idealna upotreba za *Flexbox* je kada imate skup predmeta i želite ih ravnomjerno izlagati u spremniku. Dopustite veličini sadržaja da odluči koliko individualnog prostor svaka stavka zauzima. Ako predmeti prelaze na novu liniju, *Flexbox* će razraditi njihov razmak na temelju njihove veličine i raspoloživog prostora na toj liniji, te ih adekvatno raspodijeli (Borgen, 2017).

CSS grid je namijenjen rasporedu (engl. *layout*). Kada koristimo *CSS Grid*, stvaramo raspored i stavljamo stavke u njega ili dopuštamo automatsko postavljanje stavke u mrežne ćelije prema striktnom rasporedu (Borgen, 2017).

5.2. CSS Grid raspored kao zamjena za Bootstrap layout

Bootstrap framework, je skup alata koji služi za stvaranje stranica. On koristi svoju *grid* verziju koja je zadužena za raspored stranice. Upravo ono za što se koristi i *CSS Grid*. *Bootstrap* postoji dugi niz godina, i za to se vrijeme promaknuo u jedno od vodećih

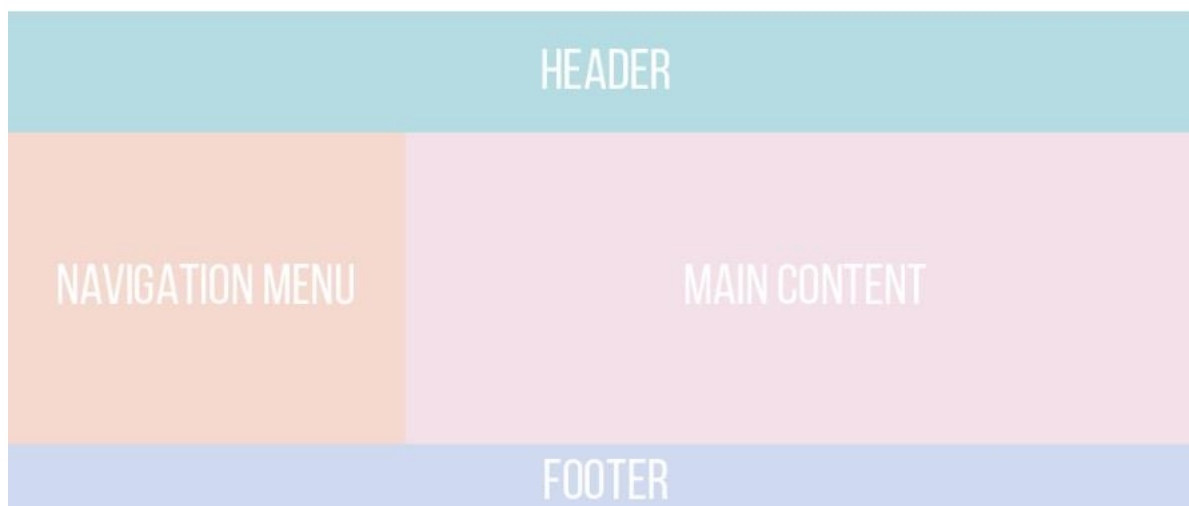
programskih rješenja za stvaranje mrežne stranice. Zbog opće prisutnosti i kasne implementacije *CSS grid*-a, *Bootstrap* je i dalje omiljeni alat za stvaranje rasporeda mrežnih stranica.

Bootstrap grid je vjerojatno jedan od najvažnijih aspekata *Bootstrap frameworka*. On je temelj na kojem se stvara cijeli raspored. Osim toga, *Bootstrap*-ov osnovni paket *CSS*-a će također dodati korisne stilove na obrasce, tablice, gumbe, popise i slike, kao i potpuno funkcionalne navigacijske trake.

Upravo količina podatak koju *Bootstrap* generira čini *CSS grid* superiornijim odabirom, govorimo li samo o izradi rasporeda (engl. *layout*) stranice. Zamjena *Bootstrapa* s *CSS Gridom* učinit će *HTML* preglednijim. Iako to nije najvažnija korist, vjerojatno je prva koju ćete primijetiti (Borgen, 2017).

Mijenjanje rasporeda također je lakše učiniti pomoću *CSS grid*-a, *Bootstrap* se oslanja na stvaranje elemente (`<div>` ili ``) unutar *HTML* koda kako bi lakše upravljao. *CSS grid* koristi isključivo *CSS*, potreban mu je samo jedan spremnik koji sadrži *grid*. Sav sadržaj može se premještati neovisno o tome gdje se unutar *HTML* kod-a nalazi sadržaj.

Utvrđujem da je *CSS grid* definitivno prilagodljiviji, lakše se njime rukuje i kao najvažnije obilježje, najlakše ga se mijenja (Kui, 2018). Kako bi prikazao razlike između *CSS grid*-a i *Bootstrapa* unutar *HTML*-a napraviti ću primjer jednostavnog rasporeda mrežne stranice vidljivog na slici 8.



Slika 8 Prikaz jednostavnog rasporeda stranice Preuzeto s: <https://blog.theodo.fr/2018/03/stop-using-bootstrap-layout-thanks-to-css-grid/>

```
1 <body>
2   <div class="container">
3     <div class="header">...</div>
4     <div class="navigation-menu">...</div>
5     <div class="main-content">...</div>
6     <div class="footer">...</div>
7   </div>
8 </body>
```

Slika 9 Prikaz HTMLkoda koristeći CSS grid ,potrebnog za jednostavni raspored mrežne stranice

```
1 <body>
2   <div class="container">
3     <div class="row">
4       <div class="col-12 header">...</div>
5     </div>
6     <div class="row">
7       <div class="col-4 navigation-menu">...</div>
8       <div class="col-8 main-content">...</div>
9     </div>
10    <div class="row">
11      <div class="col-12 footer">...</div>
12    </div>
13  </div>
14 </body>
```

Slika 10 Prikaz HTMLkoda koristeći Bootstrap,potrebnog za jednostavni raspored mrežne stranice

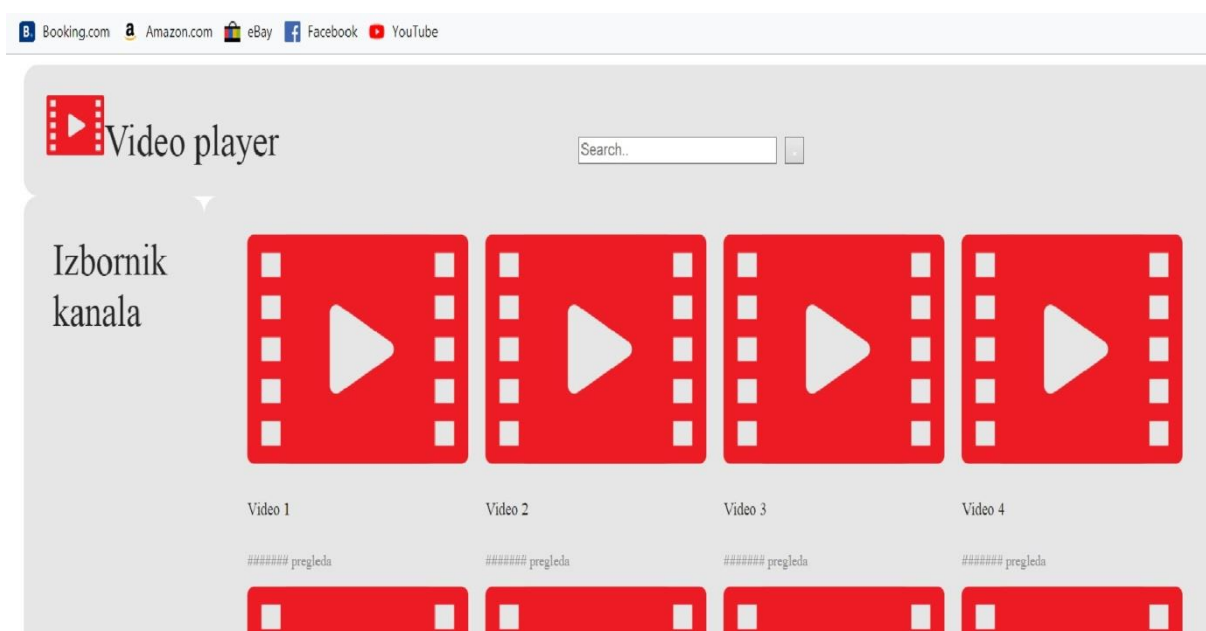
10, Bootstrap za izradu jednostavnog rasporeda mrežne stranice zahtjeva puno više kodnih redova, posebno se ističe broj <div> oznaka. Kod CSS *grid*-a ističe se prije spomenuta preglednost koda.

6. Stvaranje rasporeda postojećih web stranica uz pomoć CSS-grida

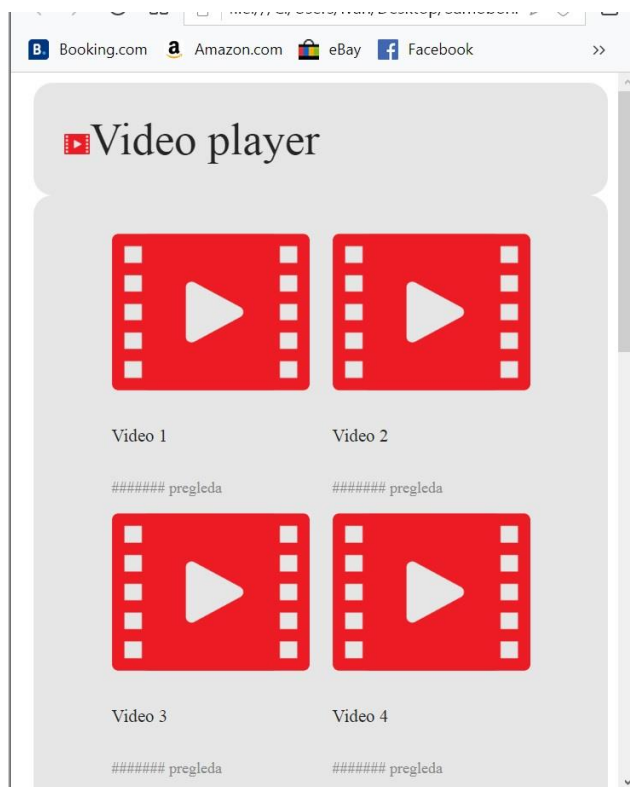
Ovo poglavlje posvećeno je primjerima izrađenima samo uz pomoć CSS grid-a. Kako bih prikazao lakoću kojom CSS grid generira raspored mrežnih stranica, izradio sam dva primjera mrežnih stranica. Prvi se primjer prikazuje mrežnu stranicu po uzoru na YouTube.

6.1. Dinamični raspored mrežne stranice po uzoru na YouTube u CSS grid-u

Ovaj kod napravljen je po uzoru na primjer Cameron Jenkinson-a (2017).



Slika 11 - Prikaz rasporeda mrežne stranice na rezoluciji osobnog računala



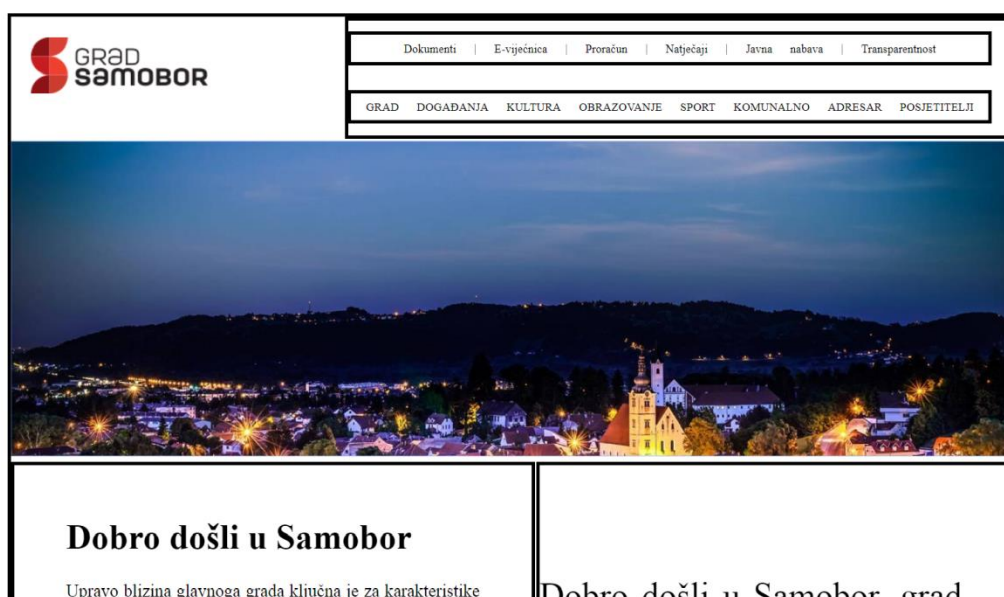
Slika 12 - Prikaz rasporeda mrežne stranice na rezoluciji mobilnog uređaja

Kao što je vidljivo na slici 12, raspored mrežne stranice na rezoluciji mobilnog uređaja daje redove i stupce na ekranu. Iz kodnog bloka u **prilogu 1** vidljivo je da se stranica sastoji isključivo od zasebnog dokumenta *HTML* unutar kojega se nalaze samo *HTML* oznake i zasebnog *CSS* dokumenta. Time smo očuvali urednost i čitkost *HTML* dokumenta (vidi Prilog 1).

6.2.

Raspored mrežne stranice grada Samobora u CSS grid-

Mrežna stranica grada Samobora idealna je za repliciranje koristeći *CSS grid*. Sami dizajn stranice kao da je zamišljen za *CSS-grid*. Stranica je izrađena pomoću *Bootstrapa*. Odabrao sam stranicu dobrodošlice (slika 13) koja je dostupna na [https://www.samobor.hr/grad/dobro-dosli-u-samobor-c13\(7.9.2018\)](https://www.samobor.hr/grad/dobro-dosli-u-samobor-c13(7.9.2018))



Slika 13- prikaz mrežne stranice dobrodošlice grada Samobora u CSS Gridu

Kako bih bolje prikazao dijelove CSS grid-au ovome sam primjeru unutar CSS koda ostavio svojstvo border na svako područje unutar grida(vidljivo na slici 13).

Na primjeru mrežne stranice grada Samobora lako je vidljiv veliki broj <div> koje *Bootstrap* zahtjeva kako bi oblikovao sadržaj. Za razliku od toga CSS grid zahtjeva samo HTML spremnik, što je vidljivo u kodu (vidi Prilog 2).

7. Zaključak

U ovom sam radu želio prikazati mogućnosti novo implementiranog alata za web dizajn i tlocrt web stranica za kojeg postoji veliki interes zajednice web programera. Predstavivši povijest *CSS grid*-a htio sam ukazati na potrebu i podršku zajednice web programera za stvaranjem *grid* alata, te što je sprječavalo njegov razvoj i čijim je zaslugama ipak implementiran. U radu sam opisao što sve treba za početak rada s *CSS grid*-om. Koji su bitni elementi za pravilno rukovanje *CSS grid*-om, te kako ih pravilno koristiti.

Predočivši mogućnosti *CSS grid*-a, te usporedbom s *CSS Flexbox*-om i *Bootstrapom*, prikazao sam zašto se upravo njega može smatrati velikim pomakom u web dizajnu. Na temelju primjera, prikazana je jednostavnost kojim pomoću *CSS grid*-a stvaramo raspored mrežne stranice. Rasporedi koji su do sada bili ovisni o svojstvima *float* i *align*, trebali bi nestati.

Smatram da je budućnost *CSS grid*-a s ljudima kao što su Jen Simmons, Rachel Andrew i Morten Rand-Hendriksen sigurna. Njihov trud i ideje garantiraju daljnje razvijanje *CSS grid*-a. Stvaranje alata za stiliziranja *grid* ćelija i područja već je u planu. Pojavljuju razni editori koji počinju olakšavati rad s *grid*-om. Očekujem, revoluciju u *web* dizajnu, potaknutu upravo *CSS grid*-om.

8. Literatura

1. Andrew, Rachel (2015): „Three years with CSS Grid Layout“ Dohvaćeno s <https://rachelandrew.co.uk/archives/2015/11/03/three-years-with-css-grid-layout/> (15.8.2018)
2. Andrew, Rachel(2012): „Giving Content Priority with CSS3 Grid Layout“ Dohvaćeno s: <https://24ways.org/2012/css3-grid-layout/> (5.8.2018)
3. Borgen, Per Harald(2017a): “The ultimate CSS battle: Grid vs Flexbox“ Dohvaćeno s: <https://hackernoon.com/the-ultimate-css-battle-grid-vs-flexbox-d40da0449faf> (8.9.2018)
4. Borgen, Per Harald(2017b): „Why CSS Grid is better than Bootstrap for creating layouts“ Dohvaćeno s: <https://hackernoon.com/how-css-grid-beats-bootstrap-85d5881cf163>(8.9.2018)
5. Bos, Bert: CSS Template Layout Module:W3 CSS Template Layout Module
6. Coyier, Chris (2018): “A Complete Guide to Flexbox“ Dohvaćeno s: <https://css-tricks.com/snippets/css/a-guide-to-flexbox/> (7.9.2018)
7. Cupp,Phil; Glazman,Daniel;Mielke,Markus, Mogilevsky,Alex: “Grid Layout:W3C Working Draft 7 April 2011“(2011) Dohvaćeno s: <https://www.w3.org/TR/2011/WD-css3-grid-layout-20110407/> (7.8.2018)
8. Gustafson, Aaron (2017): „The Story of CSS Grid, from Its Creators“ Dohvaćeno s: <https://alistapart.com/article/the-story-of-css-grid-from-its-creators> (7.8.2018)
9. House, Chris(2018): “A Complete Guide to Grid“ Dohvaćeno s: <https://css-tricks.com/snippets/css/complete-guide-grid/> (7.9.2018)
10. Jenkinson, Cameron (2017): “Learn CSS Grid by recreating familiar layouts: Airbnb, Youtube and more” Dohvaćeno s: <https://medium.freecodecamp.org/learning-css-grid-through-recreating-airbnb-youtube-and-more-399c71377eaa> (9.9.2018)
11. Kui, Cédric(2018): “How I stopped using Bootstrap’s layout thanks to CSS Grid“ Dohvaćeno s: <https://blog.theodo.fr/2018/03/stop-using-bootstrap-layout-thanks-to-css-grid/> (9.9.2018)
12. Meyer, Eric A.(2016): “Grid Layout in CSS Interface Layout for the Web“

13. Ruluks, Sandijs (2014): „A brief history of web design for designers“ Dohvaćeno s:
<http://blog.froont.com/brief-history-of-web-design-for-designers/> (5.8.2018)
14. Soares, Will(2018): ”Why you should already be using CSS Grid” Dohvaćeno s:
<https://dev.to/willamesoares/why-you-should-already-be-using-css-grid-pch> (9.9.2018)

9. Prilozi

9.1. Prilog 1 Kodni blok mrežne stranice po uzoru na Youtube u CSS Grid-u

10. HTML

```
1  <!DOCTYPE html>
2  <html>
3
4  <head>
5    <link rel="stylesheet" href="Samobor.css">
6  </head>
7  <body>
8
9    <div class="wrapper">
10     <header class="header">
11       Video player
12       <input type="text" class="search" placeholder="Search..">
13       <button class="submit" type="submit"></button>
14     </header>
15
16     <aside class="izbornik">Izbornik kanala</aside>
17
18     <div class="content">
19
20       <div class="video">
21         
22         <span class="video-title">Video 1</span>
23         <br>
24         <span class="video-subtitle">##### pregleda</span>
25       </div>
26
27       <div class="video">
28         
29         <span class="video-title">Video 2</span>
30         <br>
31         <span class="video-subtitle">##### pregleda</span>
32       </div>
33
34       <div class="video">
35         
36         <span class="video-title">Video 3</span>
37         <br>
38         <span class="video-subtitle">##### pregleda</span>
39       </div>
40     </div>
41   </div>
42 </body>
43 </html>
```

```
41 <div class="video">
42 
43 <span class="video-title">Video 4</span>
44 <br>
45 <span class="video-subtitle">##### pregleda</span>
46 </div>
47
48 <div class="video">
49 
50 <span class="video-title">Video 5</span>
51 <br>
52 <span class="video-subtitle">##### pregleda</span>
53 </div>
54
55 <div class="video">
56 
57 <span class="video-title">Video 6</span>
58 <br>
59 <span class="video-subtitle">##### pregleda</span>
60 </div>
61 |
62 <div class="video">
63 
64 <span class="video-title">Video 7</span>
65 <br>
66 <span class="video-subtitle">##### pregleda</span>
67 </div>
68
69 <div class="video">
70 
71 <span class="video-title">Video 8</span>
72 <br>
73 <span class="video-subtitle">##### pregleda</span>
74 </div>
75 <div class="video">
76 
77 <span class="video-title">Video 9</span>
78 <br>
79 <span class="video-subtitle">##### pregleda</span>
80 </div>
81
82 <div class="video">
83 
84 <span class="video-title">Video 10</span>
85 <br>
86 <span class="video-subtitle">##### pregleda</span>
87 </div>
88
89 <div class="video">
90 
91 <span class="video-title">Video 11</span>
92 <br>
93 <span class="video-subtitle">##### pregleda</span>
94 </div>
95
96 <div class="video">
97 
98 <span class="video-title">Video 12</span>
99 <br>
100 <span class="video-subtitle">##### pregleda</span>
101 </div>
102 </div>
103 </div>
```

```
1 ▾ body {  
2     font-family: "Times New Roman", Times, serif;  
3     background-color: #fff;  
4 }  
5 ▾ .logo {  
6     width: 5%;  
7     height: 5%;  
8 }  
9 ▾ .wrapper {  
10    margin-left: 10px;  
11    display: grid;  
12    grid-template-columns: 15fr 85fr;  
13    grid-template-rows: 1fr 1fr;  
14 }  
15 ▾ .search {  
16    margin-left: 300px;  
17    width: 200px;  
18 }  
19 ▾ .searchicon {  
20    width: 3px;  
21    height: 3px;  
22 }  
23 ▾ .wrapper > * {  
24    background-color: #e6e6e6;  
25    color: #242424;  
26    padding: 30px;  
27    border: 10px;  
28    border-radius: 15px;  
29    font-size: 35px;  
30    margin-bottom: 10px;  
31 }  
32  
33 ▾ .header {  
34    padding: 24px;  
35 }  
36  
37 ▾ .header, .footer {  
38    grid-column: 1 / -1;  
39    clear: both;  
40 }
```

II. CSS

```
41 ▾ .content {  
42    padding-right: 40px;  
43    padding-left: 40px;  
44    display: grid;  
45    margin: 0 auto;  
46    grid-template-columns: repeat(auto-fill, minmax(200px, 1fr));  
47    grid-auto-rows: minmax(150px, auto);  
48    grid-gap: 8px;  
49 }  
50  
51 ▾ .video {  
52    margin-left: 5px;  
53    margin-right: 5px;  
54 }  
55  
56 ▾ .video-img {  
57    width: 100%;  
58    height: 80%;  
59    border: 10px;  
60    border-radius: 5px;  
61 }  
62  
63 ▾ .video-title {  
64    font-size: 14px;  
65 }  
66  
67 ▾ .video-subtitle {  
68    font-size: 12px;  
69    color: #888888;  
70 }  
71  
72 @media (max-width: 1200px) {  
73     ▾ .wrapper {  
74         grid-template-columns: 2fr;  
75     }  
76  
77     ▾ .izbornik {  
78         display: none;  
79     }  
80 }
```

```
80 ▾ .content {
81   width: 100%;
82   grid-template-columns: repeat(3, minmax(200px, 2fr));
83   grid-auto-rows: minmax(150px, auto);
84 }
85 }
86 ▾ @media (max-width: 900px){
87   .search{
88     display: none;
89   }
90   .submit{
91     display: none;
92   }
93 }
94 ▾ @media (max-width: 768px) {
95   .content {
96     padding-right: 48px;
97     padding-left: 48px;
98     width: 100%;
99     grid-template-columns: repeat(2, minmax(200px, 3fr));
100    grid-auto-rows: minmax(150px, auto);
101  }
102  .header, .footer {
103    grid-column: 1 / -1;
104    width: 100%;
105    clear: both;
106  }
107 }
108
109 ▾ @supports (display: grid) {
110   .wrapper > * {
111     width: auto;
112     margin: 0;
113   }
114 }
```

11.1. Prilog 2: Kod rasporeda mrežne stranice grada Samobora u CSS grid-u

12. HTML

```

1  <!DOCTYPE html>
2  <html>
3
4  <head>
5      <link rel="stylesheet" href="Samobor2.css">
6      <meta name="viewport" content="initial-scale=1">
7  </head>
8  <body>
9
10     <div class="grid-container">
11         <div class="Grad samobor">
12             
13         </div>
14         <div class="Nav">
15             <p>Dokumenti | E-vijećnica | Proračun | Natječaji | Javna nabava | Transparentnost</p><br>
16             <p class="p1">GRAD      DOGAĐANJA      KULTURA      OBRAZOVANJE      SPORT      KOMUNALNO      ADRESAR      POSJETITELJI</p>
17         </div>
18         <div class="Slika">
19             
20         </div>
21         <div class="Dobro1">
22             <h1 class="naslov">Dobro došli u Samobor</h1>
23             <p>Upravo blizina glavnoga grada ključna je za karakteristike današnjeg Samobora – modernog, razvijenog,
24                 financijski stabilnog i poželjnog za nova ulaganja. Ipak, Samobor nikad nije postao tek predgrađem glavnoga grada,
25                 već je sačuvao svoj identitet, svoje prirodne ljepote, kulturu i tradiciju.</p>
26             <p>Vođeni tom vizijom našeg grada krenuli smo i s kreiranjem nove Internet stranice Grada Samobora. Želja nam je
27                 bila stvoriti stranicu na kojoj ćete naći sve potrebne informacije iz djelokruga Gradske uprave, ali ujedno i
28                 uživati u ljepotama Samobora, kojeg još više želimo razvijati kao turističku meku kontinentalne Hrvatske.</p>
29             <p>Građani Samobora tako će na jednom mjestu pronaći sve važne dokumente i obavijesti, ali i informacije o
30                 društvenim događanjima u našem gradu, kao i galeriju fotografija u čijem stvaranju i sami sudjeluju.</p>
31             <p>Vjerujemo kako će ova stranica privući i posjetitelje našega grada i pokazati im da je Samobor puno više od
32                 svima omiljenih kremšnita i naše najpoznatije manifestacije, Samoborskog fašnika.</p>
33             <p>Nadamo se da ćemo u tome uspjeti, a Vaša pitanja, savjeti i dobronamjerne kritike i dalje će biti poticaj da u
34                 svome radu budemo još bolji, moderniji i transparentniji, a time i bliži našim građanima i posjetiteljima.</p>
35             <p>Gradonačelnik
36                 Krešo Beljak, mag.educ.geogr.</p>
37         </div>
38         <div class="Dobro2">
39             <p>Dobro došli u Samobor, grad iznimne povijesti i kulture, po svemu jedinstven i poseban, nadomak velegrada, tek
40                 dvadesetak kilometara udaljen od hrvatske prijestolnice Zagreba.</p>
41         </div>
42     </div>
43 </body>
44 </html>
45

```

```
1  ▾ .grid-container {
2      display: grid;
3      height: 100%;
4      grid-template-columns: 1.75fr 1fr 1.5fr 1fr;
5      grid-template-rows: 0,2fr 1fr 1fr;
6      grid-gap: 1px 1px;
7      grid-template-areas: ". . . ." ". . . ." ". . . .";
8      font-family: serif;
9      border: solid 3px;
10 }
11 ▾ .logo{
12     padding:20px;
13 }
14 ▾ .Grad samobor {
15     grid-area: 1 / 1 / 2 / 2;
16     padding-left: 20px;
17     border: solid 3px;
18 }
19
20 ▾ .Nav {
21     display: grid;
22     grid-area: 1 / 2 / 2 / 5;
23     grid-template-columns: 0,1fr 0fr;
24     grid-template-rows: 0,2fr 1fr;
25     grid-gap: 1px 1px;
26     grid-template-areas: ". ." ". .";
27     border: solid 3px;
28 }
29
30 ▾ .Nav > p {
31     font-size: 10px;
32     text-align: center;
33     font-size: 2vh;
34     word-spacing: 15px;
35     padding: 1%;
36     border: solid 3px;
37 }
38
39 ▾ .Slika {
40     grid-area: 2 / 1 / 3 / 5;
```

13. CSS

```
41 }
42 ▾ .panorama{
43     width:100%;
44     height:100%;
45 }
46 ▾ .Dobro1 {
47     grid-area: 3 / 1 / 4 / 3;
48     text-align: justify;
49     padding: 10%;
50     padding-top: 5%;
51     font-size: 3vh;
52     font-size-adjust: 0.58;
53     border: solid 3px;
54 }
55 ▾ .Dobro2 {
56     align-content: space-around;
57     grid-area: 3 / 3 / 4 / 5;
58     text-align: justify;
59     padding-left: 10%;
60     padding-right: 10%;
61     padding-top: 15%;
62     font-size: 4vh;
63     font-size-adjust: 0.58;
64     border: solid 3px;
65 }
66 ▾ @media (max-width: 700px){
67     ▾ .Nav{
68         display: none;
69     }
```